Development Manual

Of

PLATINE V3.0

V. Baudin

Updated 24/01/2008 15:29

Content Page

1 INTRODUCTION 5

1.1 BASIC CONCEPTS 5
1.1.1 Collaborative e-learning session 5
1.1.2 Asynchronous and synchronous phases 5
1.1.3 User roles 7
1.1.4 Floor control within a synchronous phase 8
1.1.5 Flexibility of the approach 8
1.2 COLLABORATION MODEL 9

2 COLLABORATIVE PLATFORM ARCHITECTURE 10

3 GENERIC PLATFORM TOOLS 12

3.1 ADMINISTRATIVE SUPPORT TOOLS 12
3.1.1 Off-line session tools 12
3.1.2 On-line session tools 13
3.2 ASYNCHRONOUS TOOLS 13
3.3 SYNCHRONOUS TOOLS 14
3.3.1 Informal communication among users 14
3.3.2 Document sharing 14
3.3.3 Multi-user applications 15

ANNEX E – PLATINE LICENCE 18

1 Introduction

PLATINE is a collaborative work platform which has been developed in the Laboratory for Analysis and Architecture of Systems (LAAS) over the past 8 years. It allows actors of different domains, to work in a collaborative way despite their geographical environmental location. PLATINE has been used in the domain of E-learning and in particular, in the European project Lab@Future (http://www.labfuture.net/showcase/)
The purpose of this document is to present some basic concepts used by the PLATINE platform, and to explain how the whole system must be installed. Others documents have been already written for users (administrators of sessions, and participants for some sessions) All those information are available from the PLATINE web site: http://www.laas.fr/PLATINE .
PLATINE V3.0 available under the CeCILL-B licence. (see E annex)

1.1 Basic concepts

The purpose of the collaborative platform is to provide all the required communication and collaboration facilities among students and teachers, within a distributed setting, to satisfy the requirements defined by the pedagogic theories for each experiment. To perform distant access to learning experiments, the platform includes a general purpose collaborative platform linked to an experiment specific platform, using specific protocol defined with the Lab@Future project.

1.1.1 Collaborative e-learning session

A collaborative e-learning session is constituted by a group of persons working collaboratively, handling experiment specific data, performing experiment specific applications and using a set of groupware tools as support of their work. The groupware tools provide the basic communication and collaboration services to the users registered in the session.

The size of a session is assumed to be about *ten* persons. As a consequence, the network infrastructure and the server platforms are to be tuned to support the collaborative scenarios for groups of that size.

The collaborative platform may run several concurrent collaborative sessions, which are completely independent.

A session is created, configured and deleted by an *administrator*. An administrator is a specific user who logs into the system with a particular password (there may be either one single password for the collaborative platform, or a password per type of experiment).

1.1.2 Asynchronous and synchronous phases

Two phases are defined within a session, called respectively an *asynchronous* phase and a *synchronous* phase. When a session is created, it enters an asynchronous phase that is continuously active until the session is deleted by the administrator.

In the asynchronous phase, the registered users are authorized to access independently (i.e. without any synchronization among them) the e-learning content of an experiment (for instance, access to experiment specific Web pages) and to perform independently experiment

specific applications. During the asynchronous phase, only asynchronous communication tools, like E-mail, or external communication tools (like POTS) are available to the users.

The administrator is responsible to start and end manually a synchronous phase.

Once a synchronous phase is running, it is the responsibility of each user to explicitly join/leave this synchronous phase. As a consequence, all the users, who have joined a synchronous phase of a session, are aware of each other, and may use different synchronous collaboration and communication tools to work together and implement for example a pedagogic scenario.

It is important to notice that the asynchronous and synchronous phases are not mutually exclusive and may coexist. That is, when a synchronous phase starts, the asynchronous phase keeps active. In this way, users are always allowed to work asynchronously while taking also part in a synchronous activity. Thus, during the lifetime of a session, there are one asynchronous phase and possibly several successive synchronous phases (at any time there is at most one active synchronous phase).

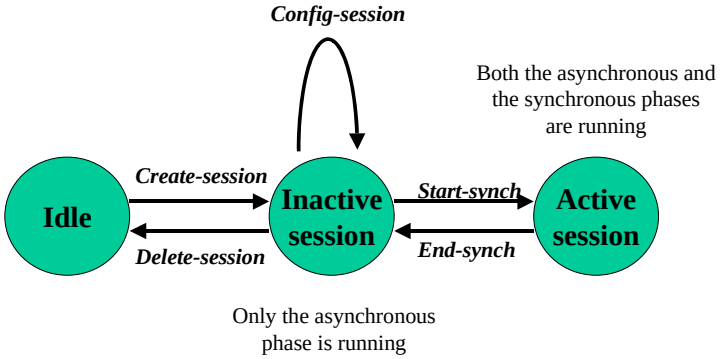


Figure 1. *Control states of a collaborative session*

We say that a session is *active* if its synchronous phase has been activated by the administrator, otherwise it is said to be *inactive* (i.e. only the tools and data defined for the asynchronous phase are available to the registered users). Figure 1 depicts a state machine representing the control states of a collaborative session, with respect to the interactions of the administrator .

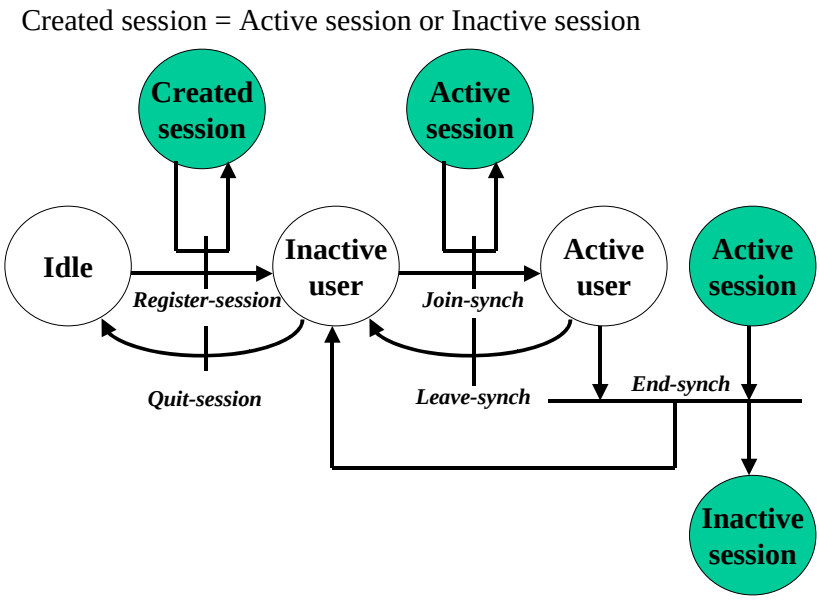


Figure 2. *Control states of a user within a session*

We say that a user is *active* if he/she joined an active session, otherwise he/she is said to be *inactive*. Figure 2 depicts a Petri net representing the control states of a user, with respect to the interactions of this user and to one interaction of the administrator.

1.1.3 User roles

Besides the *administrator* role which is associated with the user in charge of the management of the session (creation/deletion, configuration of the session), four additional user roles have been defined, which are respectively: *teacher*, *student*, *observer* and *expert*.

Thus, a user, when registering in a session, authenticates him/herself with a specific role. This role determines the rights that the user will have for using the collaboration and communication tools available in the session.

The *teacher* and *student* roles are self-explanatory.

The *expert* role is intended to implement some kind of help desk. It is detained by the users in charge of the problems and questions related to the use of the collaborative platform and/or to experiment specific applications. The help provided by an expert is restricted to the use of the general-purpose collaboration and communication tools and of the experiment specific applications. Other problems related to network configuration and system management are assumed to be out of the scope of the expert and will be supported by standard procedures (i.e. not explicitly supported and controlled by the platform).

The *observer* role is intended to represent a degraded role of the student one. Typically, an observer will have a restricted access to the tools of the collaborative platform. This may be motivated by two main reasons:

- Political reasons related to the session management implying that some users are not allowed to use the full functionality of the platform

- Network QoS constraints implying that some distant users are not capable for technical reasons to use the full functionality of the platform.

The association between user roles and collaboration/communication rights are completely defined by the administrator when configuring the session. They may be modified during the lifetime of a session.

1.1.4 Floor control within a synchronous phase

Another important issue concerns the particular problem of who is in charge of controlling the floor in an active session .

The floor represents a temporary permission to access and manipulate the shared resources. A typical solution to the floor control is the *chaired-based control*, where a human facilitator is in charge of explicitly granting and releasing floor. In the context of this platform, the shared resources are the different collaboration tools that can be executed during the synchronous phase. Then, in this case there is always a central chairman who dynamically grants the floor of each collaboration tool to a different user. For example, suppose that for *tool A*, the chairman grants the floor to *user X*, while for *tool B*, he grants its floor to *user Y*. This means that at this moment, *user X* is the only one allowed to manipulate *tool A* and *user Y* is the only one allowed to manipulate *tool B*.

In order to implement a *chaired-based control*, a *chairman* status has been defined, which may be assigned to one or a subgroup of roles. Persons that are associated with these roles and are active users are authorized to request the chairman status and to potentially gain it. However, this mechanism cannot guarantee the presence of a chairman during the synchronous phase. For example, persons who are authorized to request the chairman status have not yet registered, are not yet active or, if active, have not yet requested the chairman status.

To ensure the presence of a chairman an automaton has been implemented. When the session is started, this automaton is activated and, through a predefined scheduling, it makes the expected floor assignments for the active collaboration tools. Once an authorized user requests the chairman status, the automaton is deactivated and the respective user starts being the session chairman, until he decides not to do anymore. At this moment the automaton is then reactivated.

If an authorized user wishes to request the chairman status while another one holds it, they might use some sort of “social” protocol to negotiate the status switching.

1.1.5 Flexibility of the approach

User roles are roles and not persons. A person can log either with a role or another depending on his knowledge of the required passwords. The unique restriction is that a person connected to the platform through some workstation may always play one and only one role at a time. This means that a person may log, for instance, as administrator and then as teacher and back as administrator, but he cannot behave at the same time as administrator and teacher. However, nothing avoids the same person to connect as teacher from a workstation and as administrator from another workstation. In this case, the same person will concurrently act as a teacher and as an administrator.

There is no a priori restriction on the number of teachers, students, experts and observers who may be connected at a time, unless explicitly specified at the level of the session configuration. Roles are thus defined once by the administrator when creating a new session, and they will not be modified during the session life time. The association between user roles and collaboration/communication rights as well as the assignment of the chairman status are also defined by the administrator, but, contrary to the roles, they may be re-configured by the administrator between consecutive synchronous phases (see Figure 1). Thus some user playing some specific role (for instance, a student) might log a session synchronous phase and not another one, or vice-versa, depending of the current synchronous phase configuration

The execution of a collaboration scenario corresponds therefore to the execution of a session synchronous phase. Switching from one scenario to another at run-time implies a reconfiguration to be made by the administrator. That means that the previous current synchronous phase has to be terminated, the session be re-configured and a new synchronous phase be started.

1.2 Collaboration model

To represent the way users of a collaborative session may exchange information among them, a formal model based on graphs has been introduced. This model is simple enough to be understood and handled in a natural way. It is also general enough to specify different collaborative schemes. The structure of a collaboration scheme is defined by a directed graph, called the collaboration graph, as shown in Figure 3. Vertices represent the members of the collaborative group, and edges the relations between them. Each user within the group is assumed to own a set of information. An edge from user “U1” to user “U2” means that “U1” owns information, which is transmitted to “U2”. Following this approach, “U1” collaborates with “U2” by sending some information to him. Ownership of information still remains to “U1”.

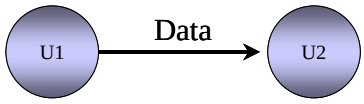


Figure 3. *. Graph-based collaboration model*

The proposed collaboration model is based on information producer/consumer relationships, to represent and process information exchanges for synchronous collaborative sessions. Such sessions handle continuous data streams (e.g. video, audio), which are specified in a natural way by specific labels associated with the edges of the model.

The application associated with a collaborative group has to define the different group structures that have a meaning for the implementation and the realization of the collaborative work. The collaborative work might be split in separate scenarios, each one being represented by a collaboration graph. Figure 4 presents two examples of scenarios modeled by two collaboration graphs.

The current group configuration evolves from one scenario to another when some specific event occurs within the group.

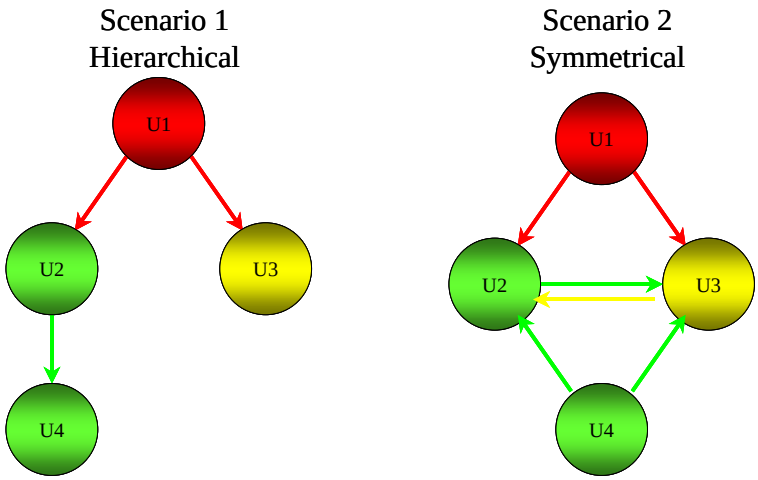


Figure 4. *Scenarios and their associated collaboration graphs*

To improve the model readability, the graph nodes are colored according to roles associated with them.

The following conventions have been defined:

- Red is associated with the teacher role
- Orange is associated with the expert role
- Green is associated with the student role
- Yellow is associated with the observer role

Arrows from a colored node are currently drawn with the same color as the node, in order to facilitate the identification of the data flow senders.

The collaboration model is used below to characterize the relationships and data exchanges among the members of e-learning sessions.

2 Collaborative platform architecture

The general architecture of the collaborative platform is presented in Figure 5. Main components of this architecture are:

- a set of distributed user workstations interconnected to a WAN network; mobile user workstations, like wireless (802.11 and possibly GPRS) laptops and PDAs, will also be considered;
- a set of distributed servers, four types of servers are considered:
    - The sessions server front-end
    - Experiment specific servers (ESS)
    - Generic collaboration and communication servers (GCCS)

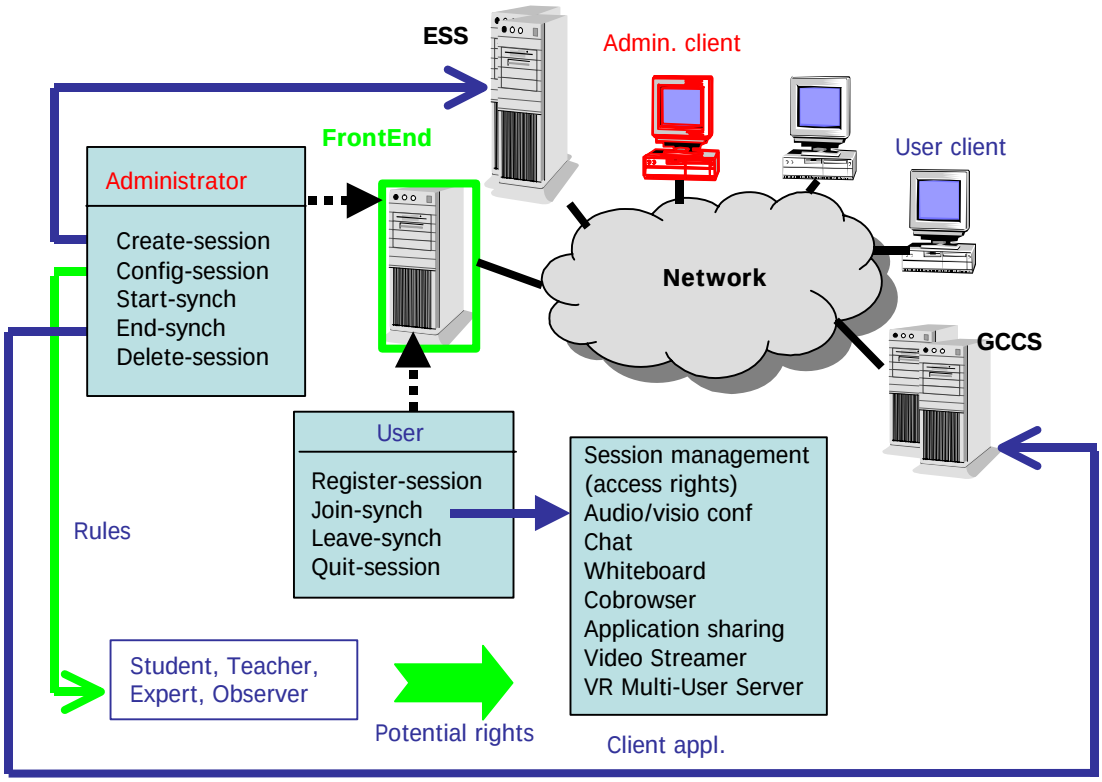


Figure 5. *Architecture of the PLATINE platform*

The set of client workstations for a collaborative session is made of, at most one administrator workstation, and one or more user workstations. The administrator is in charge of creating/deleting and configuring the collaborative session through interactions with the sessions' server front-end. For this purpose, he will use an off-line session preparation tool further detailed in the next section.

Through interactions with the sessions' server, users authenticate themselves to register in a collaborative session with a specific role. Once registered, they automatically take part in the asynchronous phase of the session. They are also responsible to join/leave the session synchronous phase if it is currently active. When registered in a session, users may access experiment specific data (e-learning content) and perhaps perform experiment specific applications. This activity is either completely asynchronous (as part of the session asynchronous phase) or it may be synchronized with the activity of other users in the same session (as part of the synchronous phase). Furthermore, users may have access to synchronous collaboration communication tools during the synchronous phase, like audio-video conferencing, whiteboard, application sharing, etc. These collaboration and communication tools will be detailed in the next section.

The sessions' server is a dedicated front end to access the platform. It provides a common Web interface for supporting the interactions of the administrator and the clients. Users may then access experiment specific data (Web pages, …), experiment specific applications and collaboration communication tools from a Web browser supporting required plug-ins if necessary. The software tools of the generic collaboration communication servers are automatically activated from the sessions' server when the session becomes active. In a similar way, the software tools of a client workstation are automatically activated from the sessions' server when the respective user becomes active. All these software tools are also

automatically deactivated when the session becomes inactive (server and client parts) or when a client becomes inactive (client part). Furthermore, all the required client software may be downloaded and installed on the client workstation from the sessions' server. These features guarantee the flexibility and transparency of the platform.

Experiment specific servers (ESS) are dedicated data servers storing all the data and applications related to the particular experiment associated with the session.

3 Generic platform tools

The tools implemented in the collaborative platform are classified into three main categories: administrative support tools, asynchronous tools and synchronous tools.

3.1 Administrative support tools

These tools implement the functionalities required for session configuration and management. There are off-line session tools, and on-line session tools.

3.1.1 Off-line session tools

3.1.1.1 Session preparation

This tool provides the administrator with a Web interface for creating and configuring a collaborative e-learning session. The configuration of a session includes the definition of several parameters, including:

- The name of the session
- The user roles considered in the session (a non empty subset of teacher, student, observer and expert)
- The passwords required for registering in the session with a user role
- If needed, the URL of the experiment specific servers accessible to the session users
- The collaboration communication tools available in the session and the definition of their associated IP addresses (IP address of the generic collaboration communication servers, multicast IP address for audio video conferencing, …)
- If needed, the web pages or a specific application (with web access) that will be automatically opened when the user enters the asynchronous phase and/or the synchronous phase of this session
- The collaboration communication rules associated with the user roles (which user role is authorized to use which collaboration communication tool and how); this feature ensures the flexibility of the platform, since it is possible to adapt for each session the way each user of a specific role is authorized to access collaboration and communication tools
- Which subset of user roles is authorized to get the chairman status for managing the synchronous phase once started.

The session preparation tool will save all the required configuration parameters in an adequate session configuration file.

3.1.1.2 Session registration

This tool provides the users with a Web interface for registering in a session (identification and authentication by session name, user id, password related to the user role) and for quitting a session.

3.1.2 On-line session tools

3.1.2.1 Session synchronous phase management

This tool provides the administrator of a session with a Web interface for managing the start and end of a synchronous phase (in case of the session synchronous phases are managed manually). The tool is in charge of activating/deactivating the server parts of the synchronous collaboration and communication tools.

This tool provides also the users of a session with a Web interface for managing how they can join/leave a session synchronous phase. The module is in charge of activating/deactivating the client parts of the synchronous collaboration and communication tools.

3.1.2.2 Session state display

This tool is in charge of providing user awareness information by displaying the session state components. These components include:

- The name of the session
- The current status of the session (inactive or active)
- The list of the users currently present in the synchronous phase of this session with the indication of their current role (administrator, teacher, student, observer, expert)
- The identification of the user having the chairman status, if any
- The list of all the generic tools configured in the session (these tools are only activated when the session is active)
- For each active generic tool, the identification of the current user to which the floor has been granted; two distinct floors are defined:
    - The document sharing floor required to load or annotate a document with the whiteboard tool, to select and transmit a video with the video streamer
    - The application sharing floor required to interact with a shared application controlled by the application sharing server

3.2 Asynchronous Tools

These tools are used when the users are working in an autonomous way, each one performing his/her own experiment without any synchronous interaction with other users of the session (this corresponds to the session asynchronous phase). If users want to communicate with others, they use asynchronous communication tools like e-mail, or any other communication tool outside of the collaborative platform. Because they are registered in a session, the users are authorized to access experiment specific data (Web pages, VRML

scenes, …) and applications through classical Web browsers including the required plug-ins .

3.3 Synchronous Tools

Three main collaboration and communication services are available during a session synchronous phase:

- Informal communication among users through audio video-conferencing (in a multicast setting) and chat
- Document sharing through whiteboard and video streaming
- Management of multi-user applications through application sharing or multi-user virtual environments.

3.3.1 Informal communication among users

This service allows users to directly communicate with others through audio, video and text messages while the collaborative activity takes place.

3.3.1.1 Multipoint video/audio conferencing

This tool provides an audio/video informal communication service among the users. It addresses the capture, compression, transmission, decompression and presentation of the multimedia streams and it is based on real or simulated *multicast* communication.

To support real multicast, a native multicast service with an adequate level of QoS, mostly in term of bandwidth requirement, must be available. The videoconference interface is composed of two types of video windows:

- A local video window, captured by the local video camera, monitors the images sent to the users; this window can be hidden to save space on the screen
- The remote video windows display the images from the remote users; they are dynamically updated according to the synchronous group structure modifications.

To ensure compatibility with other current videoconference systems, the video and audio streams are based on the H.263 standard format.

3.3.1.2 Chat

This tool provides a textual informal communication service among users. It can be used in parallel with the video/audio conferencing tool or as a backup informal communication service when the video/audio conferencing tool encounters problems, because it is more robust and less bandwidth consuming.

3.3.2 Document sharing

This service allows users to concurrently look at the same document and possibly to annotate them in a controlled way.

3.3.2.1 Shared whiteboard

This tool provides a graphical document sharing service among the users. It reproduces the behavior of a classical whiteboard in a distributed way. The interface to the shared whiteboard is composed of a graphical window for each user. An authorized user (the one who get the floor) loads images on the whiteboard (for example, JPEG, GIF images) and writes annotations on them. These images and annotations are sent to the other users (the whiteboard clients). These annotations can be saved locally in association with the appropriate images.

3.3.2.2 Video streamer

This tool provides a video document sharing service among the users. It makes it possible to broadcast a previously stored video (audio and video) file in a streaming mode. The tool may also be used to broadcast live video; in this latter case, the camera used by the video streamer tool should not be the same as the one used by the video conferencing tool. (not available for instance)

3.3.3 Multi-user applications

This third service consists of two client-server environments that have in common the ability to allow multiple users to run the same application and to synchronously interact with it.

3.3.3.1 Application sharing

This tool provides users with a generic service for sharing any application. Two different points of view may be considered:

- Any specific application shared during a session is stored on a server; all the users see the execution of the application; an user may interact with the application according to the access right granted by the chairman
- Each user owns an instance of the shared application, with his/her own data; according to the access right managed by the chairman, other users can see and possibly access (in a remote control mode) this application.

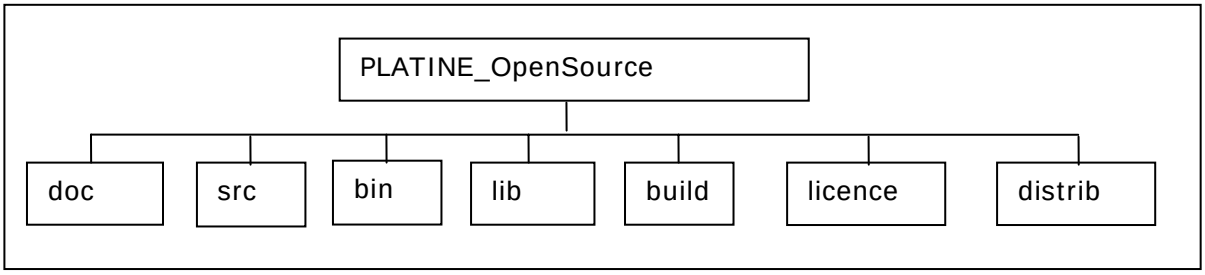
In both cases, the client side of the application sharing tool represents a remote view of the shared application. Through the client interface, users can request and, if accepted, accomplish the remote control of the shared application.

The application sharing tool periodically captures and samples the screen dumps of the shared application. They are displayed on a specific video window for each user. It may be then considered as a generic synchronous multi-user tool.

3.3.3.2 Multi-user virtual environments

Some experiment-specific applications provide users with collaborative interaction on the same virtual scene. In order to allow the sharing of interactive virtual scenes, a multi-user virtual environment can be used. This environment is not a part of the PLATINE platform.

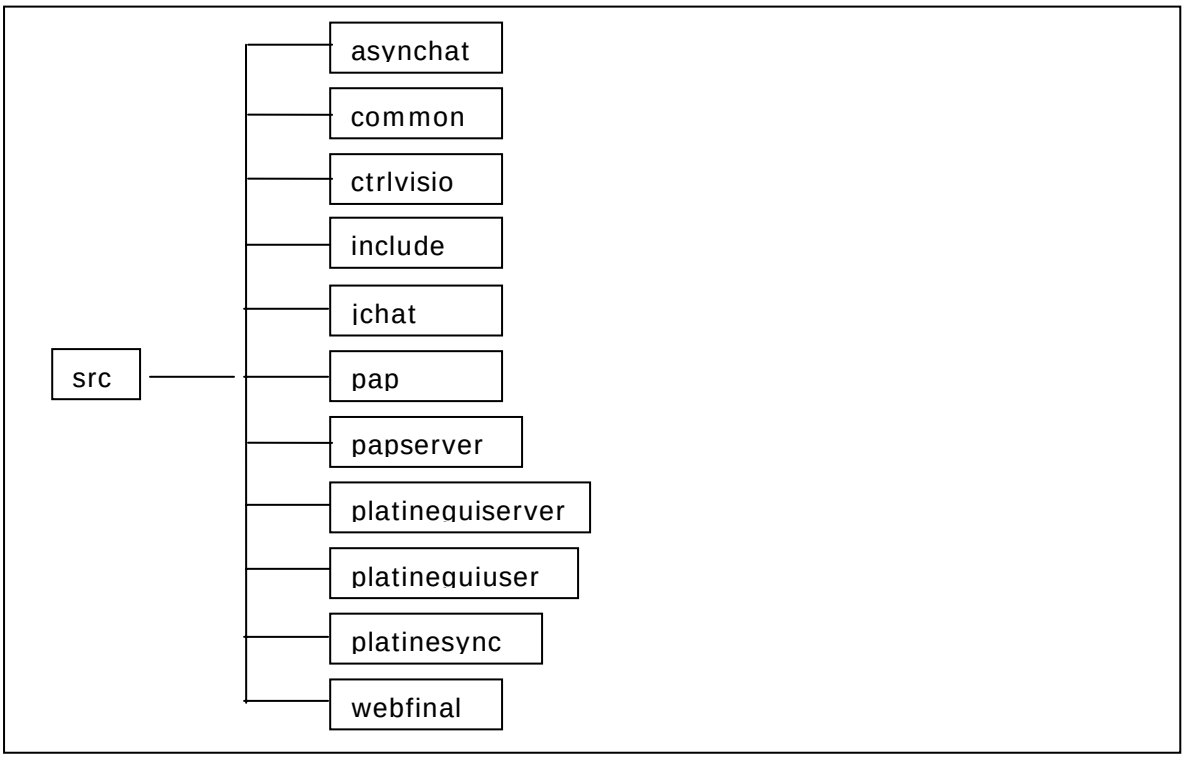
4 PLATINE tree



doc: contains the 2 guides for install and dev
src: contains all the source code of PLATINE
bin: is used to store class files when the jar files are built
lib: contains 2 directories
JMF_win: with sound and jmf jarfiles for Windows
javaws-1_2-dev: with jnlp.jar needed for some PLATINE jar files

build: contains compile scripts for both windows and linux (must be changed in ant file)
licence: contains the CeCILL-B licence used for PLATINE
distrib: contains the directory frondEnd with the needed files to start the sessions' Platine server and all the jar files needed to use PLATINE as client (administrator or user), the server.jnlp file to start PLATINE using javaWebStart, and an example of a define session called Meeting_Test in the experiment MEET, and the configuration files (LabFuture, LabFuture_Config, LabFuture_Passwords)

4.1 PLATINE/src tree



The papserver directory contains just the compiled C++ VNC server for windows in an old version: this must be changed ASAP.

Annex A – PLATINE licence

PLATINE is distributed under the CeCILL-B licence (http://www.cecill.info/index.en.html) Each PLATINE's file contains the following texte:

Copyright LAAS-CNRS
contributors : V. Baudin, M Diaz, P Owezarski, T Villemur
September, 29 2005

e-mail: veronique.baudin@laas.fr

This software is a computer program whose purpose is to support synchronous cooperative work in a group of users.

This software is governed by the CeCILL-B license under French law and abiding by the rules of distribution of free software. You can use, modify and/ or redistribute the software under the terms of the CeCILL-B license as circulated by CEA, CNRS and INRIA at the following URL "http://www.cecill.info".

As a counterpart to the access to the source code and rights to copy, modify and redistribute granted by the license, users are provided only with a limited warranty and the software's author, the holder of the economic rights, and the successive licensors have only limited liability.

In this respect, the user's attention is drawn to the risks associated with loading, using, modifying and/or developing or reproducing the software by the user in light of its specific status of free software, that may mean that it is complicated to manipulate, and that also therefore means that it is reserved for developers and experienced professionals having in-depth computer knowledge. Users are therefore encouraged to load and test the software's suitability as regards their requirements in conditions enabling the security of their systems and/or data to be ensured and, more generally, to use and operate it in the same conditions as regards security.

The fact that you are presently reading this means that you have had knowledge of the CeCILL-B license and that you accept its terms.